Initial Results of the Software-driven Navigation for Station Experiment

Evan J. Anzalone*, Chris Becker, and William H. Sims
NASA Marshall Space Flight Center, Huntsville, AL, 35812

To enable the next generation of robotic and human exploration of the solar system, improvements are needed to enable robust and accurate autonomous navigation. The purpose of this work is to take advantage of the growth in and use of software-defined platforms to incorporate additional navigation capability on existing assets, while also incorporating with new vehicle designs. The Software-driven Navigation for Station Experiment focuses on implementing two soft solutions to this: transmitting pseudolite signals to perform ranging and Doppler measurements as part of the signal coding (similar to underlying Global Navigation Satellite System approaches), and the Multi-spacecraft Autonomous Positioning System, which uses existing communication protocols to embed navigation and timing information to be shared among all assets in a peer-to-peer network. These technologies were implemented on the SCaN Testbed onboard the International Space Station and exercised over the course of mid-June and late-July 2018. This paper will discuss the operational architecture, experiment plan, and initial results from the data collected. One of the key conclusions of this work is the strong need for stable accurate clock synchronization across the dispersed space network.

I. Nomenclature

GRC	=	Glenn Research Center
GPS	=	Global Positioning System
ISS	=	International Space Station
MSFC	=	Marshall Space Flight Center
MAPS	=	Multi-spacecraft Autonomous Positioning System
NEN	=	Near Earth Network
SCaN	=	Space Communications and Navigation
SN	=	Space Network
SDR	=	Software Defined Radio
SNS	=	Software-driven Navigation for Station
TDRS	=	Tracking and Data Relay Satellite

II. Introduction

With the expanding capability and performance of software-defined platforms, orbital assets can be upgraded to allow for additional functionality through the updating of onboard software using typical microprocessors. In addition, the increasing use of field programmable gate arrays in support of Software-Defined Radios (SDR's) allows for broad implementations of additional spacecraft capability. The Software-driven Navigation for Station experiment takes advantage of flexible hardware and software platforms to demonstrate how additional navigation capability can be embedded into an existing flight program. This experiment focuses on two specific navigation technologies with a focus on their implementation, integration with, and operation within existing large-scale space architectures. Specifically, this experiment takes advantage of the SCaN SDR Testbed, mounted externally to ISS in the bow direction, to perform a series of navigation technology demonstrations. Each of the aspects of the experiment will be described in detail in the following sections. Fig.1 below provides an overview of the individual elements of the experiment, and identifies the high level goals of transmitting pseudolite signals and MAPS packets from the orbital payload down to the ground station using real-time telemetry of ISS timing, position, and velocity.

---

*Guidance and Navigation Team Lead, Guidance, Navigation, and Mission Analysis Branch, EV42
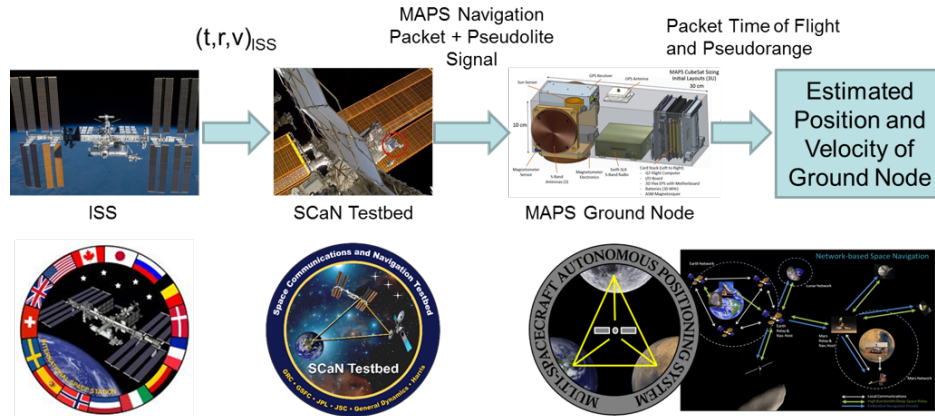
Fig. 1   High Level Experiment Architecture

The first set of experiments performed implement pseudolite[1] algorithms onto the JPL S-band radio within the Testbed. This technology has been widely used on the ground to provide GPS augmentation in areas of insufficient coverage, such as urban canyons, underground mining facilities, and areas with minimal GPS constellation coverage, such as near the poles. This technology provides a GPS-compatible signal that is transparent to the receiver and allows for the use of standard technology in state estimation. Additional applications of this technology involve their use in providing navigation support for a network of planetary rovers on Mars[2]. This first part of the experiment will broadcast pseudolite signals from the SDR Testbed down to a ground station located at Glenn Research Center. A flight-like platform will be connected to the ground station to demonstrate a low-cost, embedded pseudolite receiver. This ground hardware will measure pseudorange and Doppler of the transmitted signal and perform navigation to estimate its state (as well as to perform orbital state estimation of ISS assuming it had received the measured value instead). This part of the experiment demonstrates a well-understood technology and demonstrates its integration and application with on-orbit assets.

The second half of the experiment is an orbital demonstration of the Multi-spacecraft Autonomous Positioning Systems (MAPS). This technology represents a GPS-like approach to solar system-wide navigation by taking advantage of the growth of inter-spacecraft communications within an expanding network. To enable navigation functionality, MAPS embeds navigation-specific information in a standard format within the digital information transmitted between assets. This allows the technology to be agnostic to transmission technology, i.e. optical or RF, and communication standard by being embedded in the form of a navigation header. The participating assets are then able to form ranging and range-rate measurements by processing the timing of the received measurements through a state estimation filter. This system is heavily reliant on onboard time-keeping knowledge, but as more advanced orbital timing technology is deployed, the capability is further improved. In order to demonstrate this technology, the SDR Testbed will transmit MAPS compatible packets derived from ISS-onboard state knowledge to the ground over S-Band. Like the previous experiment, a flight-like ground node will be receiving these packets using an existing ground station at NASA Glenn Research Center. SNS will exercise the foundational MAPS software library, containing the packet generation, processing, and navigation algorithms as qualified Class-D software within a cFS architecture[3], readying MAPS for future implementation and in-space demonstration.

Another key aspect of this experiment is to assess MAPS sensitivity to the operational architecture and demonstrate the ability to calibrate an orbital platform to sufficient accuracy to support navigation functionality. In order to asses this, the MAPS packets are embedded over a series of network paths, both to allow a longer light travel baseline for measurement and to capture network latency effects. In addition to direct to ground transmission from the S-Band SDR, the MAPS packets will also be transmitted to the ground via a direct TDRS link, executing a larger communication infrastructure. This allows inclusion of a separate ground station through the use of Tracking Data and Relay Satellite (TDRS) services. Additionally, MAPS packets will also be embedded in standard SDR Testbed payload telemetry which is collected by ISS, packaged, broadcast to the ground, and provided to the experimenter over the existing ground network. This allows for transmission of MAPS packets on a variety of network interfaces of increasing complexity to provide an environment to exercise calibration routines and assess system performance.

This paper provides a description of the SDR Testbed, and its unique capabilities that have enabled this experiment.

Also presented are the individual technologies being exercised and experiment results from the flight demonstration. These results focus on initial processing of the recorded flight data to assess system stability and performance across the various transmission chains. Additionally, this research demonstrates a rapid approach to flight algorithm demonstration, through the use of an orbital asset with a well-defined and well-supported interface, that has gone from concept to flight in a year, demonstrating the strengths of the SDR Testbed and its operational approach to allow for demonstration of several navigation technologies.

A. The Multi-Spacecraft Autonomous Positioning System

The first technology being exercised is the Multi-spacecraft Autonomous Positioning System (MAPS)[4]. MAPS takes advantage of the growing inter-spacecraft communication network and infrastructure to allow for Earth-autonomous state measurements to enable network-based space navigation. A notional fully formed concept of operations for a solar system-wide GPS-like navigation network is given in Fig. 2. This network is already being implemented and routinely used in Martian communication. The growth of this communications architecture is continued through MAVEN and future potential commercial Mars telecommunication orbiters. This growing network provides an initial Mars-local capability for inter-spacecraft communication and navigation. These navigation updates are enabled by cross-communication between assets in the network, coupled with onboard navigation estimation routines to integrate packet travel time to generate ranging measurements. Inter-spacecraft communication allows for frequent state broadcasts and time updates from trusted references. The architecture is a software–based solution, enabling its implementation on a wide variety of current assets, with the operational constraints and measurement accuracy determined by existing onboard systems.
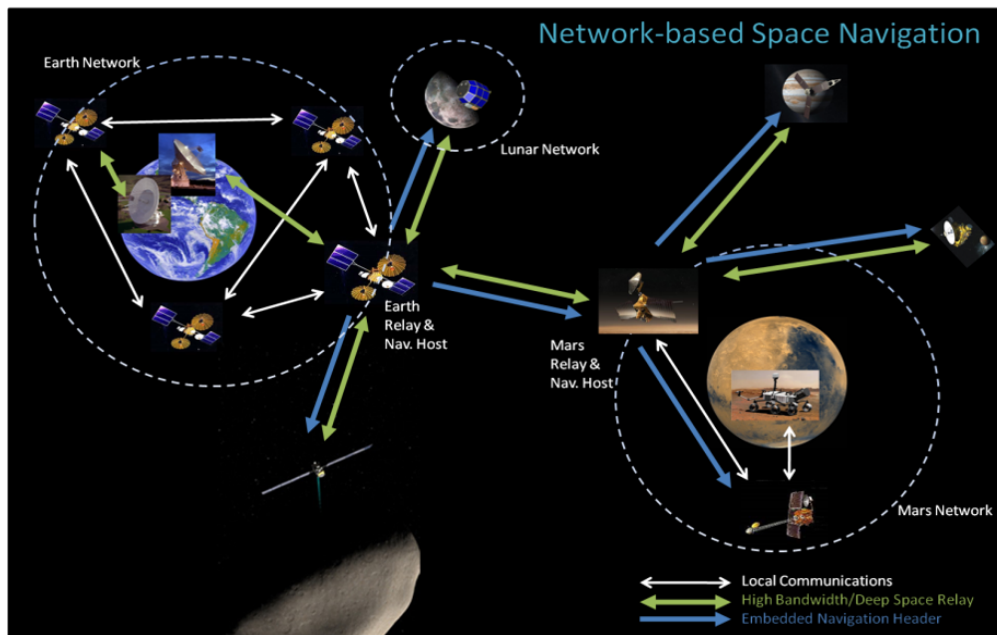

Fig. 2   MAPS Concept of Operations

This technology has been in development at MSFC for several years. Initial development resulted in a complex mutli-spacecraft simulation framework that was used to assess the capability of the architecture to support navigation for periods of deep space cruise to reduce reliance on ground tracking and to mimic a potential technology demonstration[5]. The project then moved onto hardware in the loop testing on flight-like radios using an in-house soft real-time execution framework. This testing further developed the software algorithms and provided an opportunity for initial calibration of flight-like hardware elements (such as processor grades and clocks). These tests have shown the capability of the algorithms and allowed for initial testing. In addition to the larger scale implementation, the team is also focusing on applying this technology to a local Lunar navigation network to support surface and in-orbit navigation independent of Earth networks. This concept of operations is shown in Fig. 3. Although this algorithm has been tested in ground hardware and limited testing on low altitude unmanned vehicles, this experiment will further mature the flight libraries

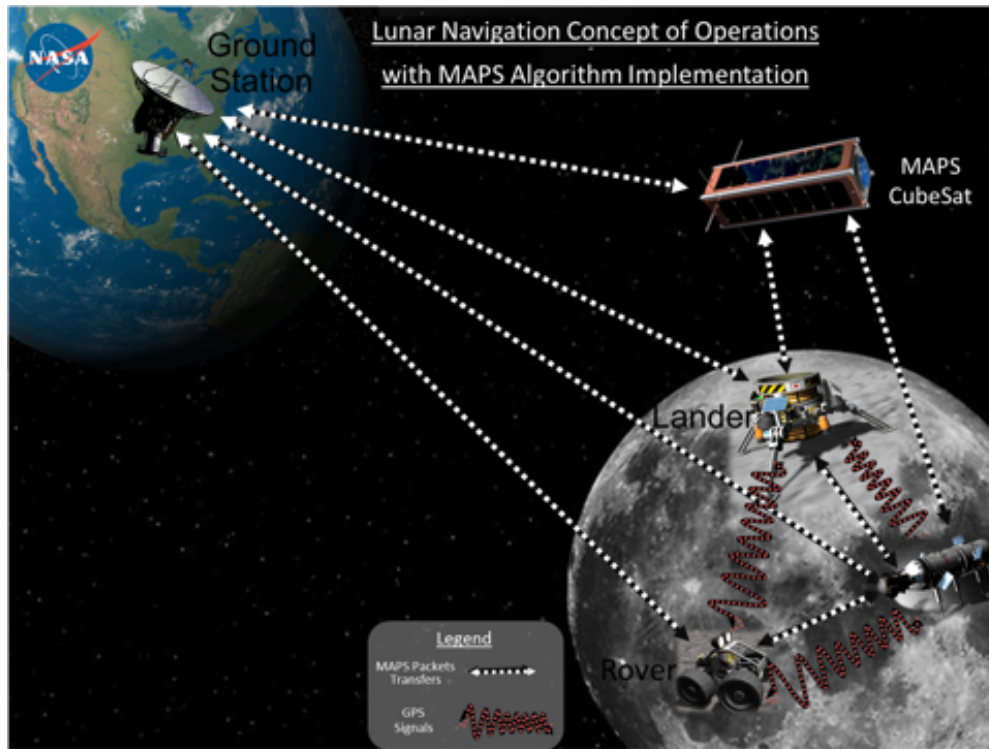and integrate this capability into an operational flight platform.

Fig. 3   MAPS Lunar-focused Concept of Operations

B. SCaN SDR Testbed

The SCaN SDR Testbed is an experimentation platform to enable research and operations of in-space radio waveform and communication technology[6][7]. To that end, the experiment itself is comprised of three separate radio platforms on different frequencies (L-band , S-band, Ku-band) as well as a centralized avionics element providing the interface to the radios and the payload commanding via ISS. The platform is installed on the forward facing truss with connections to the Express Rack interface. Fig 4 identifies the key pieces of the SDR testbed, laying out the individual assets included. As shown, the payload includes multiple antenna elements for transmission and reception as well as a steerable antenna used for communication with TDRS satellites for Ku and S-band testing and checkout. For this experiment, the JPL radio was used for waveform generation for all operation scenarios. For communication with TDRS, the experiment utilized an existing Mode F transmission[8][9] compatible with the TDRS elements in use. This mode was also used for direct communication with the ground receivers. For the pseudolite experiment, a custom waveform was developed for the JPL SDR to transmit a GPS-like waveform over S-band. This waveform simulated the Gold Code and synchronization to the local clock at the start of the UTC second on the Avionics unit. Additionally, the waveform included the capability to overlay a low rate (50 bits per second) navigation message similar to GPS operations. For this experiment, in lieu of generating standard GPS navigation message as defined in [10], MAPS packets are overlaid onto the waveform as the data message at the defined low data rate.

C. Pseudolite Applications

The SNS experiment will implement a pseudolite transmitter onboard the SDR Testbed and use it to broadcast from ISS to a specialized receiver on the ground. A pseudolite generates a signal that meets the GPS standard and is typically used to provide satellite coverage in areas with low visibility, such as urban canyons or deep underground, where GPS signals are too weak to be used. Typical applications of pseudolites in human exploration include extra-planetary navigation support, intra-vehicle relative navigation, and in-situ ground navigation systems on the Moon or Mars[11][2]. This demonstration will show how readily-available SDR's can be used to enable a new navigation capability. As part of
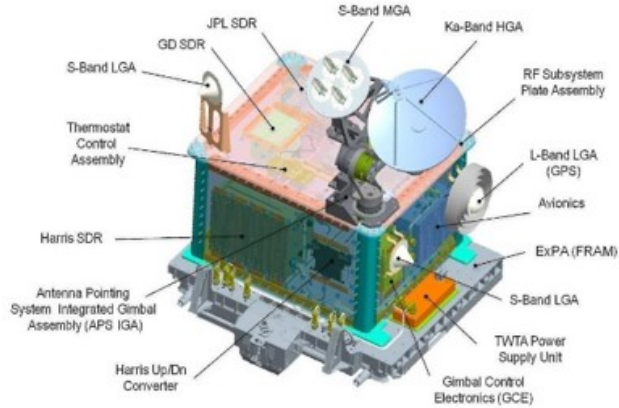
Fig. 4 SCaN SDR Testbed Components (NASA/JPL)

this demonstration, a ground node will measure range and range-rate to ISS and perform state estimation similar to a typical GPS receiver.

III. Software-driven Navigation for Station Experiment

The SNS experiment's purpose is to take advantage of a flexible orbital platform to test navigation algorithms that can be implemented solely via software integration within existing hardware assets. This use case is intended to both showcase the technology as well as demonstrate the process to go from ground hardware in the loop testing to integration into an existing vehicle. The focus of the software development was to integrate the existing C++ library (a port of original analysis performed in a Python simulation architecture [12]) into a more modular format with well defined interfaces to simplify integration into other flight systems. The following sections will provide a detailed overview of the types of experiments performed, the ground asset integration, software developed, and the operational plan in preparation for flight operations.

A. Overview of Experiment

Prior ground-based testing focused on hardware-in-the-loop testing of the algorithms within a system of flatsats. Though this approach allowed for porting of the MAPS navigation libraries into a real-time processing system with flight-like hardware, the ability to model time delays and ability to synchronize the simulation architecture limited the effectiveness of demonstrating the architecture behavior. In order to increase the Technology Readiness Level (TRL), a flight demo is needed in order to physically increase the distance between assets such that the time of flight becomes significantly larger than the inherent timing uncertainty onboard the flight hardware. Additionally, the first anticipated use case of this technology is to increase robustness of and support deep space interplanetary navigation. To this end, the team originally proposed to develop a pair of cubesats phased sufficiently to allow for large enough range between the two assets and the ISS to perform ranging among the assets with GPS being used as a truth reference. While pursuing those flight demonstration opportunities, the team considered installing an external payload on ISS dedicated to navigation experiments with measurements to an external cubesat or to the ground. In order to reduce the cost and risks, the team developed a proposal to rapidly advance the technology through collaboration with the SCaN SDR Testbed, taking advantage of the existing computational, interface, and RF transmission and reception capabilities.

A detailed concept of operations is shown in Fig. 5 below. This graphic shows the key assets participating in the experiment operations. The MAPS software will be hosted onboard the SCaN Testbed and will communicate with ISS telemetry via the Express Rack to capture current position and velocity and to provide experiment telemetry back through the ISS-to-experimenter telemetry interface. The MAPS software then integrates SDR Testbed Avionics timing information with the ISS state to develop MAPS packets to be transmitted out over RF. These packets will then travel to the ground via three operational paths: direct to the ground station at GRC where existing receivers and the ground node will receive and process the packets, over S-band to a TDRS and to GRC via operational networks, and through the ISS telemetry interface (also going through TDRS, but through a different path including processing at other NASA centers (Johnson Space Center and Marshall Space Flight Center). Additionally, the avionics software will synchronize the

generation of pseudolite signals and manage transmission of MAPS packets as the low rate navigation data message.
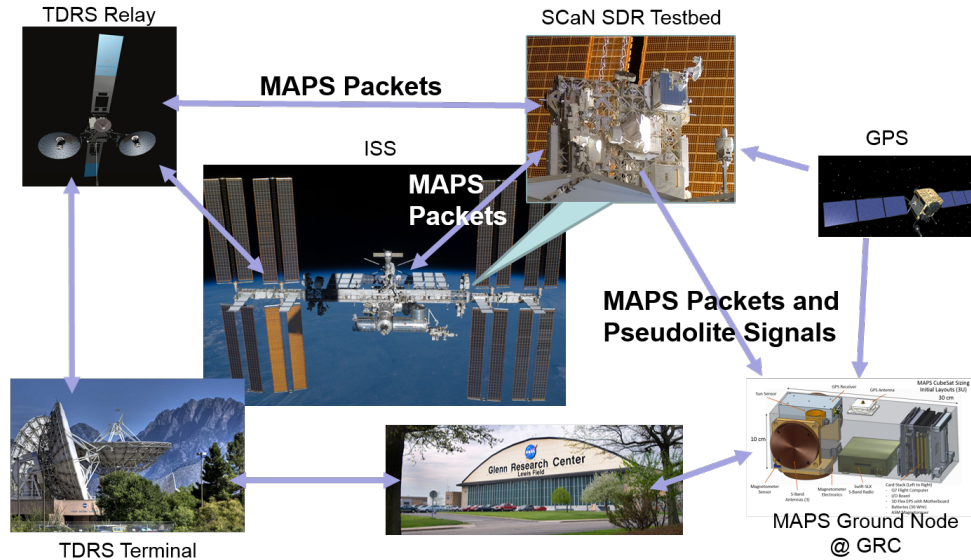

Fig. 5   SNS Concept of Operations

B. Software Development

In order to support future integration into potential flight projects and the expansions of the MAPS architecture, the MAPS algorithms were ported into specific drivers that would work within the Core Flight System (cFs)[13]. This operational approach was selected in order to reduce risk in integration of algorithms into other programs. Similarly, the ground node hardware was selected to utilize cubesat class components. The software development approach was treated similarly to developing flight software for a cubesat-scale mission. As part of this effort, the software was ranked independently by Software Quality Assurance and identified as Class D software. In order to meet organization and software best practices, NASA process 7150.2b test plans and processes were employed to meet institutional standards. The software was developed within a Linux x86 platform to enable rapid development and testing. In order to support testing on the flight platform, cross-compilers were used to prepare code for the embedded platform in the ground node during flight testing.

In order to interface with the orbital hardware, the SCaN Testbed project implemented an Experimenter's Application Interface that abstracts the interfaces to both ISS in-flight telemetry via the Express Rack interface and to the onboard avionics. Typically, the testbed hosts experimenter software in the form of waveforms on the internal SDR's. For this experiment, existing waveforms are being used (aside from the newly developed pseudolite signal generation code). The MAPS algorithms and experiment interface is being hosted by the Testbed's Avionics platform. To speed development, the team used the experimenter's interface functional definition to stub out functions. For this interface, the team developed a dedicated cFS Application that handled communication between the flight software and the underlying hardware platform. The plan for development was to port the entire cFS stack to operate on the Avionics platform, which is running a PPC604 processor architecture and running VxWorks.

The cFS software stack also interfaced with the existing C++ MAPS library to perform packet generation, processing, and state estimation. This same library was used in previous hardware-in-the-loop testing. As stated, one of the primary goals of this experiment is to wrap the MAPS algorithms into a cFS application layer. To develop this, a full suite of cFS applications were implemented to enable testing and development. The specific apps included: MAPS Navigation, Packet Processing, Telemetry Generation, Command Processing. For the ground node, additional applications were developed to interface with the Ground Radio Receiver and GPS platform. Fig. 6 shows the overall software architecture of the flight and ground software. The primary products of the software testing were a VxWorks .out object along with an Experiment specific XML file detailed experiment commands and interfaces.
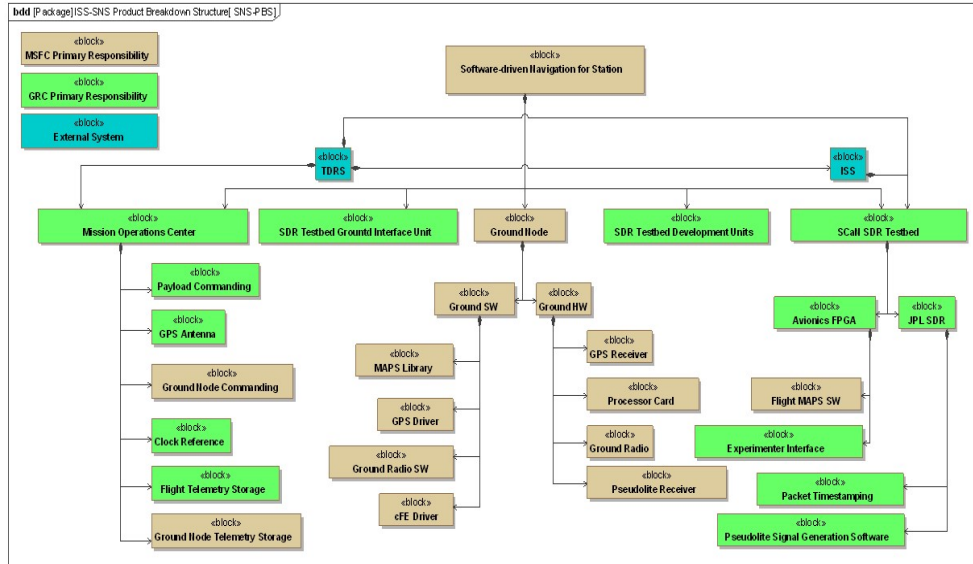
Fig. 6   Product Breakdown of Software and Hardware Elements

C. Operational Architecture

The operational architecture was spread across two buildings at GRC in support of this experiment. For reception of raw RF signals from the payload, a ground station is located on the roof of one of the buildings. This ground station is connected via hard line to the other building housing the ground hardware and Telesciences Support Center (TSC). An overview of the connected architecture is shown in the below figure for each type of pass. Each of the 3 types of unique experiment types (direct to ground, direct to TDRS (and ISS telemetry), and pseudolite) has a slightly different operational architecture in terms of elements and transmission paths. Each experiment type will be described in detail. The three types of passes are shown below in Fig. 7, which shows the active links for the various experiment types.
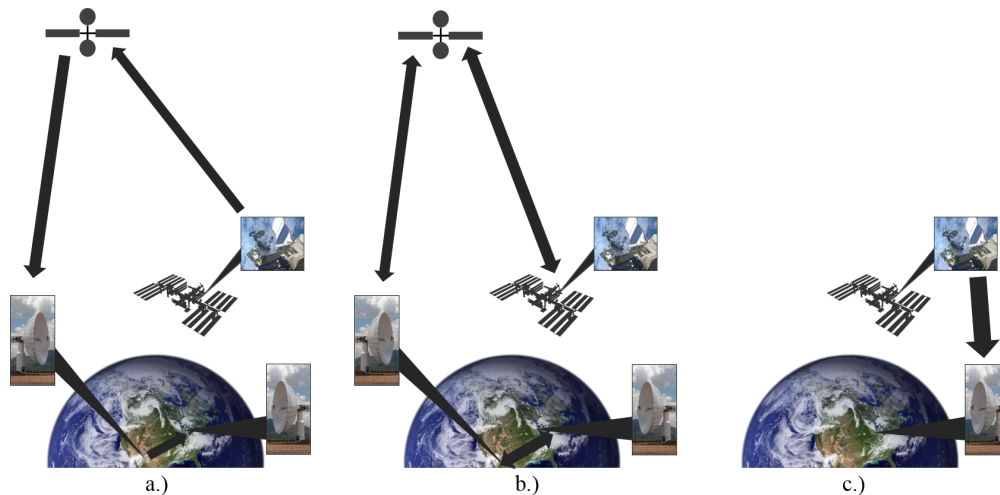

Fig. 7   Lines of Communication in Experiment
a.)MAPS Packets through TDRS b.)ISS Telemetry and Payload Commanding c.)Direct to Ground

The first type of pass utilizes a framework similar to communication with the Near Earth Network(NEN). This type of pass involves direct communication between the SCaN Testbed and the GRC ground station. The naming convention was selected to align with work previously done to enable the testbed to integrate with the NEN via the Wallops Island Ground Station. During this type of pass, the experiment software is loaded in memory, executed, and initialized by

ground operators in the TSC via ISS payload commanding functionality and the payload's Experimenter's Interface. Most passes over the ground station have a maximal duration of 6 or 7 minutes. Due to the payload's forward pointing antenna, the passes typically exhibit higher signal to noise levels as the ISS rises over the local direction. As such, having the payload in a transmission mode just prior to the beginning of a pass maximizes the ability to receive data. During this type of pass, the RF is collected by the ground station and processed by the ground receiver as well as piped to the ground node. The ground receiver decodes the transmitted packet and transmits the payload packet over a serial connection to a gateway computer. This gateway then timestamps the arrival of the packet and forwards the information (appended with reception time) to the operational network (including the ground node). The primary goal of this type of pass is to characterize the stability of ranging measurements with a minimal operational architecture between the flight and ground receiver (where the packet time of arrival is measured).

The second type of pass involves a longer baseline with a focus on navigation performance. Instead of transmitting directly to the ground, this type of pass utilizes communication with TDRS satellites in Geosynchronous Orbit (GEO). As such, the light travel distance has now increased by almost two orders of magnitude. This helps to improve the ability of the measurement itself to increase above the inherent timing stability and noise in the system. The Mode F waveform used for ground communication is also used for this type of pass. With communication to an asset in such a high orbit, the duration of the pass to a specific TDRS asset is also significantly increased. For these passes, the RF is received by the Ground Terminal operated by the Space Network (SN, which is also used to identify passes of this type), and de-modulated onsite with the serial data stream of the embedded packets transmitted to GRC via a hard line. This data is then sent to another gateway computer (similar to the one above) where the packet is timestamped and forward along to other assets on the local hardware network. With the extended baseline and increased duration of experiment events, these passes are intended to allow for data to show the navigation capability of the MAPS algorithms. The only caveat to this is the additional latency introduced by the transmission from the ground terminal to GRC. The plan for data analysis is to use some of the collected timestamp data along with best estimated trajectories to estimate the bias due to the ground network.

The last type of experiment also used direct to ground communication, but instead of mode F utilized the pseudolite waveform. The implementation of this waveform enabled external commanding to the JPL SDR to reset the phase of the transmitted code. This functionality was used prior to a pass in order to sync up the waveform as close as possible to the start of the UTC second (as defined by the clock on the SDR Testbed Avionics). Similar to the NEN passes, the RF transmission of this mode was enabled prior to the beginning of the ground to maximize amount of data received. During this pass, the RF is collected by the ground station with power reduction elements in line to offset the gain of the dish to match power levels manageable by the ground SDR. The SDR is operating GNURadio[14]-based software to attempt to measure the code phase and provide Doppler measurements to the ground node for state estimation. During these passes the pseudolite signal is continuously generated, and the avionics software provides a navigation message at a low rate consisting of a MAPS packet with onboard time and state embedded. The raw RF was also collected for future analysis and decoding.

D. SNS SW and HW Integration

As part of the experiment development process for the SCaN Testbed, any potential software must be demonstrated to cause no harm and not interfere with operations and commanding. As such, prior to uploading code to the orbital asset, all experiments must be exercised on the Ground Interface Unit (GIU) in order to gain approval of the project. This process involves loading the proposed software onto the GIU, and walking through the operational sequences and exercising all aspects of the code. A photograph of the GIU can be seen in Fig. 8 at GRC. The team spent two weeks performing software testing and integration on the ground assets with the support of the SCaN Testbed team.

The initial week of testing focused on getting the code in flight-ready configuration. As mentioned above, the software development utilized the cFS architecture and wrapped the existing MAPS library within a application. The code was developed and tested on both development machines (PCs running the cFS suite within a Linux Virtual Machine), as well as onboard the cubesat flight hardware (Xiphos Q7* running Linux). This software took advantage of the existing MAPS C++ framework to interface with the navigation algorithms. The complete suite of software used for hardware in the loop testing included variety of external support libraries such as SPICE[15], rapidJSON†, and EIGEN‡. These libraries are fully supported on Linux platforms and operated successfully. The flight platform, though, operates

Fig. 8   SCaN SDR Testbed Ground Interface Unit

using VxWorks[§] using a PPC604-type architecture. In order to meet schedule constraints, the team decoupled the flight software development from running an entire flight software stack to only running the navigation application. This required pulling the support libraries into compilation under VxWorks and the development of a streamlined application that interfaced both the SCaN experimenter software as well as the cFS application. The development of this focused implementation allowed for risk reduction and streamlined the software testing being performed on the GIU to focus on the navigation capabilities. The team was able to successfully compile the required source libraries and the streamlined application for the host platform within VxWorks.

The reduced software stack was compiled and ready for the initial testing with the GIU for the schedule integration. During testing, additional issues were identified with the support libraries included in the flight build. These missing libraries caused incompatibilities with the portions of the SNS flight software. As such, the team made a further streamlined version of the application that could operate on the payload while still maintaining the integrity of the core navigation algorithms and packet formation libraries. With these updates made, the team was able to create a version of the FSW that would operate successfully on the GIU and still meet the goals of flying the MAPS libraries. As part of the forward work, the team is working to re-integrate this version into the base MAPS library to enable a full VxWorks version of the library to be integrated onto additional flight platforms. This version of the software was tested on the GIU, and was able to properly generate MAPS packets, interface with the ISS payload telemetry stream, and successfully provide data packets to the onboard SDR.

The initial testing verified the primary functionality of the SNS experiment software. A second round of testing was required prior to flight operations in order to integrate additional ground hardware, tie into the operations network, and run through procedures with the payload operators. During this testing the ground node hardware was attached to ground network in order to receive raw RF from the locally mounted GPS receiver and ground antenna. A view of the hardware attached is shown in Fig. 9. This image highlights the Xiphos Q7 daughter board and Tethers Swift SDR. For operations the processor card was mounted onto the interface card and was also interfaced with the SpaceQuest GPS receiver shown in Fig. 10.

To enable further testing, the team also wanted to capture raw RF at the ground interface to enable continued compatibility testing and processing of the received pseudolite signals. An Intel Compute Stick was chosen as the hardware baseline to run the pseudolite data processing software and capture raw RF. This hardware interfaced with a HackRF[16] to perform RF capture and integration. Additionally, this ground hardware interface with a GPS receiver

---
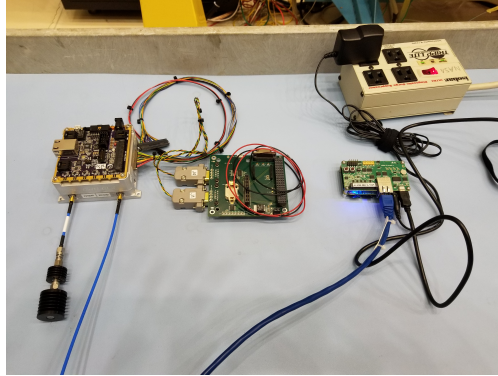
[§]https://www.windriver.com/products/vxworks/

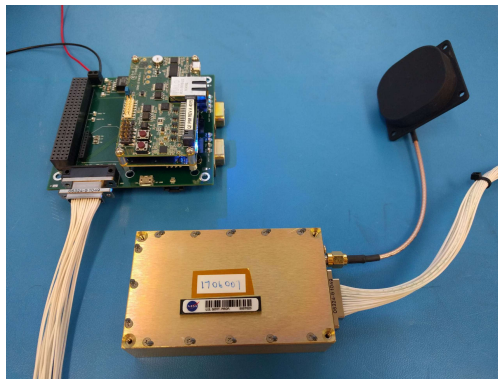**Fig. 9   SNS Ground Node Hardware**

Fig. 10   SNS Ground Node with GPS Receiver

located near the dish used for direct communication with the SDR testbed. This data was collected in order to provide a real-time measurement of the truth state of the ground station.

IV. Operational Results

As part of the experiment, two sets of operational passes were executed separated by a month. These times were selected based on availability of ground operations personnel at GRC with overhead passes during primary working hours. Both sets of passes exercised all parts of the architecture under test, including direct to ground passes and relay-based passes. The two series of passes took place June 14-22 and July 30 - August 3, 2018. SCaN Testbed operators managed the orbital payload, identification of desired ground passes, and integration with Space Network and ISS payload operations. The payload was operated out of the Telesciences Support Center at GRC. The following sections will provide additional detail into the executed operations and provide presentation and discussion of the collected data and initial results. The intent of these operations is to exercise the technologies under development and provide experience with their use in an operational environment. A particular focus is on characterizing the capabilities of the systems on as implemented hardware. The effect of network complexity will be investigated as part of the results below.

A. Executed Operations

With support from operations planners in the SCaN Testbed team at GRC, two and a half weeks of operations were scheduled. Using existing software, the planners assessed the available passes over GRC in terms of duration and power levels. This allowed for identification of ground passes that could be used as NEN or PL experiments. The team also interfaced with SN asset schedulers to determine availability to perform TDRS passes. The two data sets were then assessed for any overlaps or scheduling issues. A summary of the individual operations performed is given in Table 1. The original planning date for mid-June passes was set well in advance in order to coordinate with other testbed

Table 1   Experiment Summary

Type	Number	Max Duration (HH:MM:SS)	Mean Duration (HH:MM:SS)	Integrated Duration (HH:MM:SS)
SN	22	00:41:30	00:23:13	08:30:39
NEN	23	00:07:46	00:07:08	02:44:36
PL	11	00:08:00	00:07:09	01:18:31
All	56			12:33:46

experimenters. After the first week of passes, a second week of availability was identified. The focus of these passes was on SN-type experimenters, but several ground passes were available near the end of each day that were split between PL and NEN. Overall, the team was able to complete 56 passes and accumulate over twelve hours of operational flight time.

As shown in the table above, and mentioned previously, the NEN passes were often of much longer duration than the ground passes. Fig.11 presents a ground track of ISS during each of the experiments. Additionally, on the plots of this type, the 'x' symbol represents a ground station, while the star symbols identify the location of the TDRS relays in use. The plots show greater density over the central United States, but through the use of TDRS assets, the data collected spreads over the entire orbital plane. For each of the experiments, raw payload telemetry was recorded on the operations network as well as time updates across each of the ground assets in addition to the received packets.
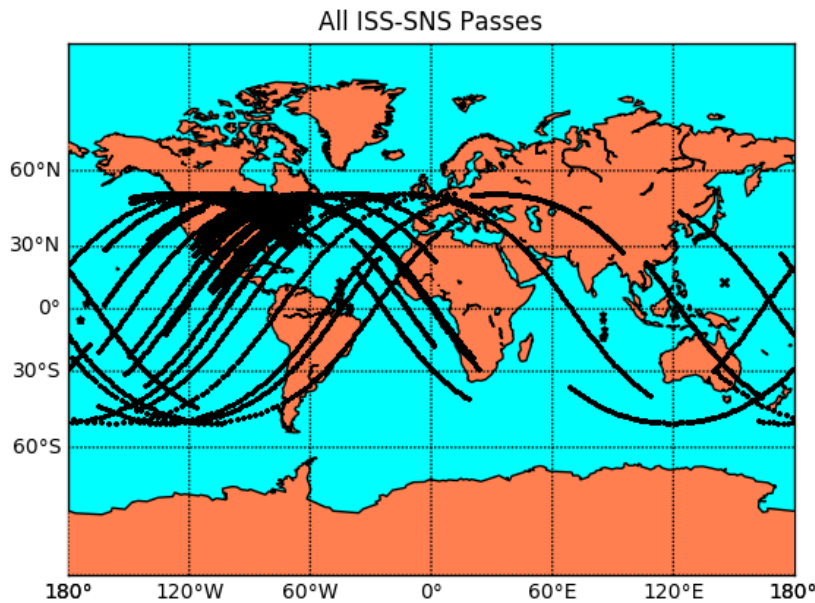

Fig. 11   Ground Track of ISS during All Experiments

B. Characterization of Ground Hardware

Prior to operational phase of the experiment, several ground characterization events were executed in order to gain insight into the underlying capability of the GIU and the ground node hardware. This includes doing timing characterization tests where Network Time Protocol logs (used for synchronizing system clocks) were used to gain insight into clock bias and drift terms. These test events also allowed testing on a system using flight radios with minimal transmission distant. One of the key tests performed was to generate packets using the onboard software, and measure the time latency of the baseline system. Fig. 12 provides data during the calibration run. First of all the negative bias of

the received time indicates that even with all systems synced together, there is a still a remaining clock bias that shows the GIU clock running a little fast. At the start of this event, all clocks were manually synced to GPS time, with the data recorder also synchronized at GPS.
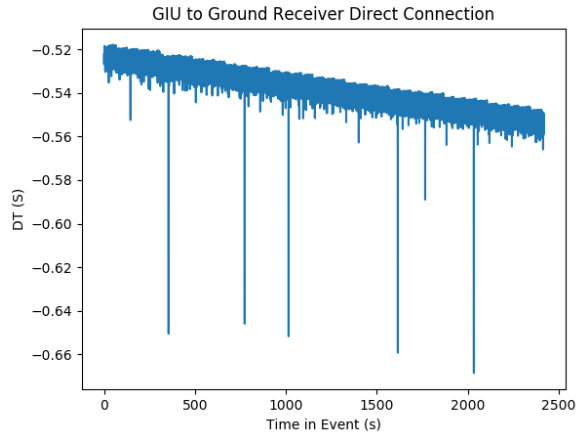

Fig. 12   DT During Calibration Testing

As shown in the figure, the short term noise is fairly constant. Interestingly, the results also show one of the clocks drifting over the course of the experiment. Additional data needs to be processed to asses the timing stability of the house platform and ground receiver during this test. This data can also be used to assess the timing stability of the ground platform. During the test the packet output rate was increased from once a second to ten times a second, similarly to the planned operations. The experiment software uses system time to meet the desired operational rate. Fig. 13 shows the stability of the receive and transmit timestamps during the ground calibration. Focusing on the data identifies an interesting quantization on the GIU's clock in that it's time between steps is quantized at 10 milliseconds. The ground node, though, exhibits a much more stable time between packets, showing the radio is outputting them at a fairly constant rate, but this does place additional uncertainty on the broadcast system timestamp.
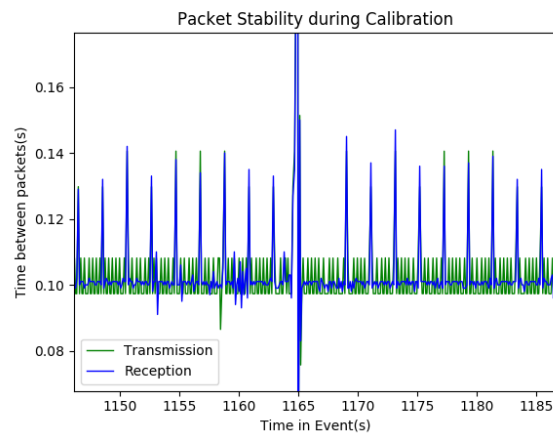

Fig. 13   Timing stability during Calibration Testing

C. Direct to Ground Navigation
The first type of pass takes advantage of the existing technology that was developed to enable the SCaN Testbed to integrate with the NEN. This type of experiment exercised direct communication between the SCaN Testbed and the ground station at GRC. As mentioned above, due to the geometry requirements the passes were typically limited to a

duration of five to seven minutes. Fig. 14 provides a plot of the ground passes that were used. The plots below show the entire time of telemetry over the pass when the experiment was online and producing telemetry received through ISS. Thanks to the second set of passes a month later, the team was able to perform operations over a range of passes in the sky.
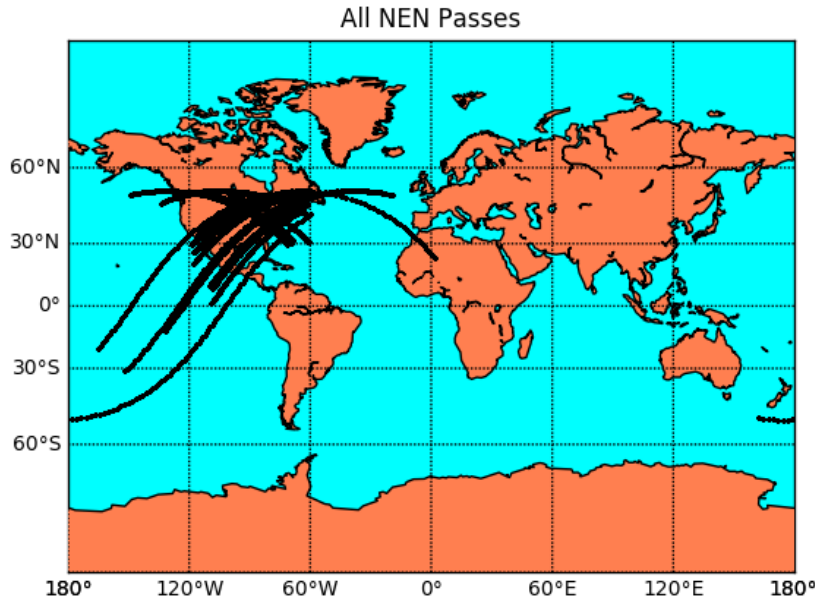

Fig. 14   Ground Track of ISS during NEN Experiments

With the ground equipment, it was possible to also observe in real-time the RF power levels. A notional output of the spectrum analyzer can be seen in Fig. 15. The ground station's dish at GRC is capable of tracking ISS as is passes overhead to maintain high signal levels and enabled a steady signal. The blue line on the graphic shows the max power level over the event, while yellow is the current signal.
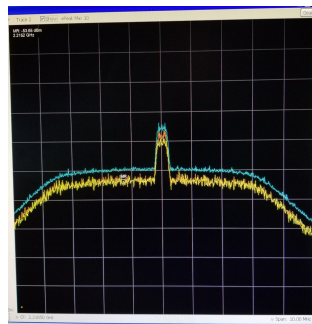
Fig. 15   Power Spectrum of Received Mode F Signal

The received RF signal was piped to a commercial ground receiver used in GIU hardware-in-the-loop testing to demodulate the transmitted data and provide a data stream to the ground node. A SDR was also used to record the raw signal for future testing and integration with software implemented receivers. Upon reception of a decoded packet on the ground system, the time of arrival was captured and used in navigation analysis. This data was decoded in real-time and output to the operator to indicate successful packet processing and the current range measurement. The raw plot of one of theses passes is given in Fig.16 below. The range was calculated by the difference between the time of transmission and time of the reception. The negative value here indicates that the ground and onboard clocks are out of sync.
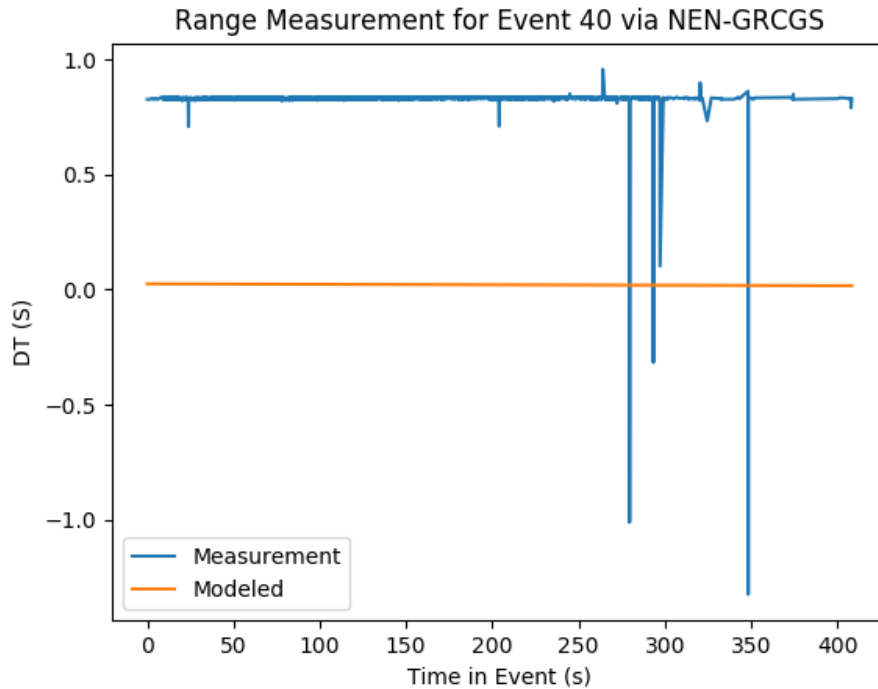
13

Fig. 16   Measured and Modeled Range across a NEN Pass

A majority of the ground passes indicated this behavior. This is due to either an offset/timing assumption mismatch in terms of leap seconds or the clocks are out of sync. In addition to understanding the source of this error, it is possible to use the data to capture a measurement of the latency and delta between clocks over the pass. Though the clocks exhibit a delta between the expected and measured range, these can be calibrated out of the system (and would be for a more tightly integrated flight system). The navigation performance is more tied to the timing instability and noise over the short term, for example if the clock noise is on the level of the measurement, it will be difficult to obtain an accurate range observation to used in vehicle state estimation. The team is actively processing the data to analyze the short term stability of the clocks to asses its capability for navigation.

D. Extended Baseline Navigation

Over the course of operations, the experiment was able to execute a large number of ground passes through the TDRS network as shown in Table 1. These passes allow for a much longer transmission baseline of the range measurement, representing a path from ISS to TDRS and back down to the ground terminal. The high altitude of the TDRS satellites enables the range measurement to increase above the noise levels during a pass. Fig. 17 demonstrates that the overall passes were able to encompass the ISS's motion across its entire orbit. These passes ranged in duration from 5 minutes up to 40 minutes. During the pass events, the output rate of packets was increased from 1 to 10 HZ to allow for a greater density of observations over a pass. The experiment also was able to utilize passes to multiple TDRS satellites. With this data, latencies will be assessed across satellites and ground stations (i.e. the transmission of data through the Guam Remote Ground Station in addition to White Sands.

Fig. 18 shows data from a notional TDRS event with data being transmitted from the SCaN Testbed to a singular satellite and through the ground network. Through assessment of the passes, different latencies have been observed depending on ground station. This is as expected due to one of the TDRS satellites providing RF capability via the Guam Remote Ground Station, incorporating additional delay between the receive station and the White Sands Facility requiring transmission of digital data over a significant distance. The plot in Fig. 18 shows a pass over a satellite in communication with White Sands. The plot identifies a timing offset in the onboard timing of a significant quantity, explaining the time of transmission being after the time of reception. This is inherent as well in the NEN passes,
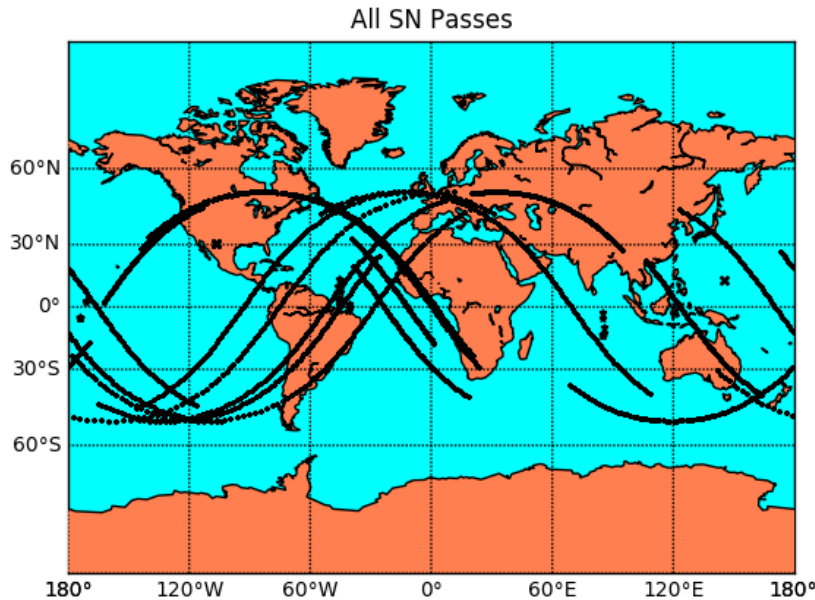
Fig. 17 Ground Track of ISS during SN Experiments

indicative of an offset on the onboard timing data used, potentially a leap second missing in the data processing scripts, or simply a floating clock that is out of sync. With this observed behavior, each pass will require calibration to estimate the clock bias during the pass in order to demonstrate the effect of timing instability over the course of an event.

It is possible to also use data from the ISS telemetry to capture latencies between individual satellites for the extended baseline. Due to the independent scheduling of ISS telemetry with TDRS, it is possible for one SN pass to encompass ISS communicating with multiple TDRS asset. Initial results of this data show measured and modeled ranges from ISS to the ground via multiple TDRS relays over a long event. For these scenarios measured range can be determined by the timestamp between transmission and reception of telemetry. This use case also represents the most complex operation architecture. In this communication path, telemetry is generated by the payload at 1 Hz, transferred to ISS, broadcast to TDRS, transferred to the ISS Ground network, and parsed through the central payload operations center at MSFC, and finally received at the TSC and timestamped on arrival. This also though best represents a fully implemented communication architecture for a human spaceflight mission. Fig. 19 captures a notional pass of ranging using ISS telemetry. In this plot, each color represents a different TDRS relay. As the data shows that there appears to be a fairly fixed latency between each of the satellites truth range and that measured. In this architecture additionally, all assets are on a common timing network, which could explain the reduced observed latency. Also, this data shows how the observed range dynamics appear to match the satellite. This type of behavior is enabled by having communication link with multiple TDRS satellites during an event. This data will be useful in characterizing expected network delays and help to differentiate it from direct TDRS observations.

E. Pseudolite Navigation

As described above, the typical passes for the PL were around seven minutes long. Due to the limited maturity of the pseudolite receiver software, these experiments focused on capturing raw received RF signals over the course of multiple pass geometries. This data will be used to continue to refine and demonstrate the phase and Doppler measurement capabilities. As shown below in Fig.20, the experiments represent a variety of flight paths over the ground sight. The ground system also captured a live power spectrum of the received signal. Fig.21 shows a characterization of the signal as received during one of the PL passes. As shown, the spread spectrum behavior the coded signal can be seen in the power distribution.
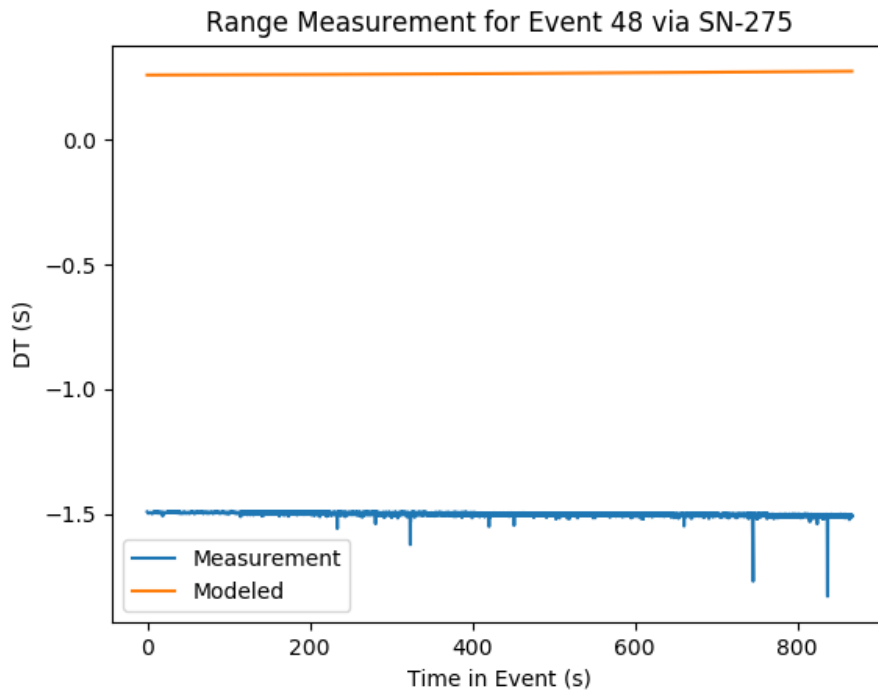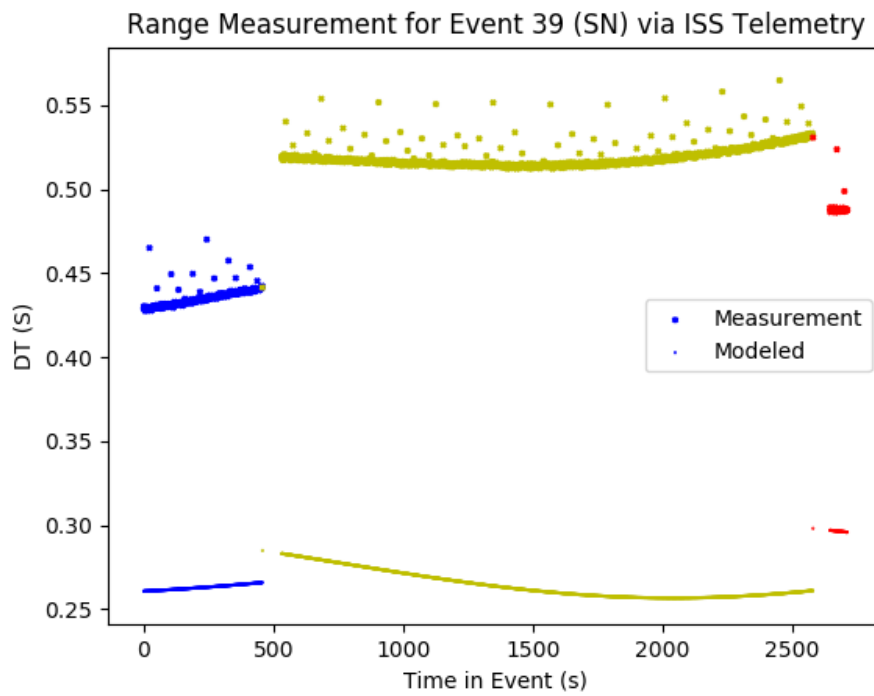
Fig. 18   Measured and Modeled Range across a SN Pass

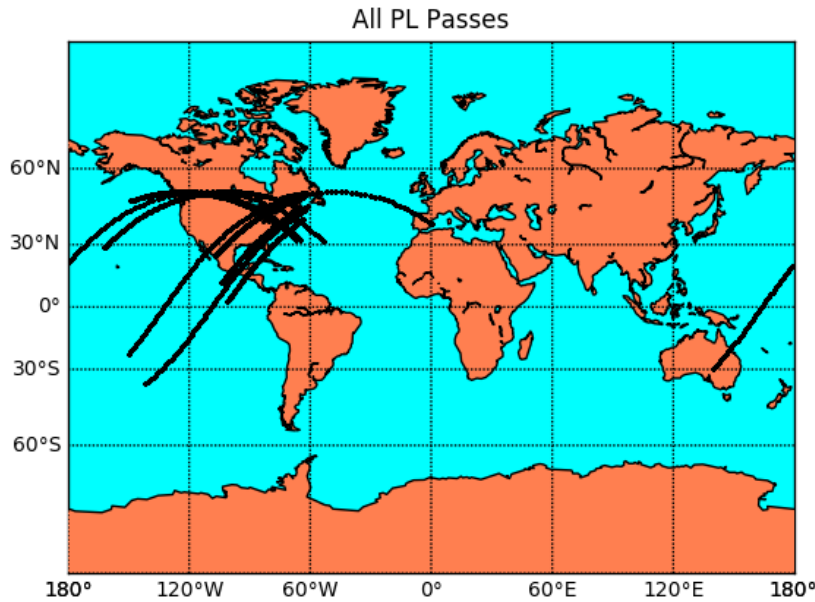**Fig. 19   Measured and Modeled Range across Multiple TDRS Satellites**

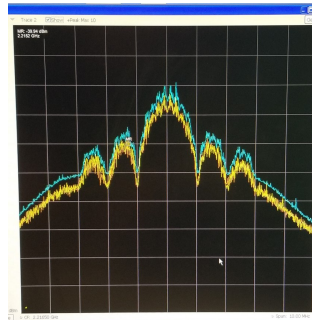**Fig. 20   Ground Track of ISS during PL Experiments**

Fig. 21   Power Spectrum of Received Pseudolite Signal

F. Forward Analysis Plan

In order to meet the hard constraint on the operations schedule, the experiment had to diverge slightly from its initial goals of having a cubesat platform fully integrated into the ground station at GRC. The team, though, was able to meet the core goals of demonstrating the MAPS-specific navigation library as integrated and executed on a host platform in an orbital environment. The team was able to successfully make ranging measurements over a large number of pass. The focus of the continuing work is to continue to process the received data to assess timing stability and begin to run the measurements through orbital determination routines to determine state uncertainty. For the plots presented here, publicly available Two Line Elements were used to generate as expected flight trajectories. The team is working to get higher quality post-processed trajectory data to better assess measurement accuracy.

Similarly, the team will continue to integrate and test the hardware developed. This includes using other ground terminal simulators to modulate packets similar to that used during the experiment. With this hardware, the team will continue to do hardware in the loop testing of the full cFS software stack. This will enable completion of radio compatibility testing, GPS integration, and execution of the algorithms in real-time on the cubesat platform.

V. Conclusion

This experiment demonstrated an approach to test navigation algorithms on an orbital platform in a very quick turnaround. The existing plans and agreements centered around the SCaN Testbed Experimenter's Interface allowed for rapid development, testing, and upload of flight software. If these had not been in place, the software certification alone (being onboard a human spaceflight system) would have drawn out the project significantly. As this project was rapidly executed with a very small team, this rapid, existing process towards software certification and testing are key enablers. These enabled the project management to make smart decisions in regards of tuning systems engineering processes to meet tight deadlines in regards to on-orbit scheduling on the SCaN SDR Testbed.

Taking advantage of an existing payload in orbit allowed the team to focus on software development. Similarly, the availability of ground receivers as part of the operational network at GRC allowed the team to focus on software development of the flight and ground algorithms to meet the fundamental goals of demonstrating the core navigation algorithms. While this approach did enable a rapid turnaround on the experiment development from proposal to flight (one year), it also introduced some limitations on how well integrated the experiment software could be. This limited the accuracy of the exact measurements, principally due to the existing limitations on timing synchronizations between ISS and the SDR Testbed Avionics as well as between the SDR Testbed Avionics and the JPL SDR Platform. This limited the accuracy of the onboard timing solution. Similarly, taking advantage of the existing ground hardware, while enabling a rapid path to flight, also introduced its own inherent latency. Due to the limited scope and schedule of the experiment, it was not able to complete full characterization of the ground networks that were integrated as part of the longer baseline experiment, principally the ground links between GRC and WSC. This timing uncertainty places limitations on the final accuracy of the navigation algorithms. These limitations reduce the ability to demonstrate the performance of the navigation algorithms immediately, but the team is developing methods to use the data to calibrate ground delays to help characterize the systems. At the time of this writing, the real-time truth data is still being generated and integrated into analysis tools for this purpose.

One of the primary limitations of this experiment is due to existing time synchronization algorithms onboard the payload. The Avionics platform performs a time sync with ISS every ten seconds. There is some internal latency due to the decoupling of the radio from the avionics and the setup of the waveform. The waveform continually sends out idle (or empty) data frames until data is ready for transmit. The size of the data packet produces an uncertainty in the delta between packet transmission from avionics and radio frequency output. This resulting quantization-like effect is easily seen in the results above. For a future experiment, detailed calibration needs to be performed to understand and characterize the behavior of the radio (something the team will complete with the receiver on hand to implement and test calibration processes). Additionally, timestamping the packet in the radio as part of the signal processing would reduce the latency assuming the radios onboard clock was well-synced. These lessons provide necessary insight into future development efforts.

The team is continuing to process the data and demonstrate navigation potential. Similarly, the team will continue to mature and perform hardware in the loop testing with the cubesat flatsat hardware. This testing will help to continue develop calibration algorithms, demonstrate navigation performance, and enable testing of the flight radios in the loop. Integration with the flight radios in a cross-link configuration will enable calibration of radio-specific latency. Similarly, ground receiver simulations will also be used to characterize their latency, providing more insight into the flight data. Additionally, the team is continuing to develop the pseudolite receiver algorithms to decode and improve the accuracy of the range and range-rate measurements from the raw RF data collected during the flight passes.

The operational architecture and rapid schedule to meet flight deadlines did limit performance demonstrations during the flight passes. Even so, the team was able to collect a wealth of data and experience in terms of how to integrate the MAPS algorithms into a flight platform. The data collected also provides significant insight into the effects of the growing ground architecture and its effect on measurement stability and latency. The core purpose of this experiment was to port the algorithms to a flight architecture and capture its performance. With the development of the flight software within the cFS architecture and operational experience and lessons learned, the team has the experience and insight to support future collaborations and continued development of this technology.

Acknowledgments

References

[1] Cobb, H. S., *GPS pseudolites: Theory, design, and applications*, Stanford University, Stanford, CA,, USA, 1997.

[2] LeMaster, E. A., and Rock, S. M., "A local-area GPS pseudolite-based Mars Navigation System," *IEEE 10th International Conference on Advanced Robotics, Budapest, Hungary*, 2001.

[3] McComas, D., "NASA/GSFC's Flight Software Core Flight System," 2012.

[4] Anzalone, E., and Chuang, J., "Conceptual Design of a Communication-based Deep Space Navigation Network," *AIAA SPACE 2012 Conference & Exposition*, 2012, p. 5186.

[5] Anzalone, E., Becker, C., Crump, D., and Heater, D., "Multi-spacecraft Autonomous Positioning System: LEO Demo Development," *AIAA/USU Small Satellite Conference*, 2015.

[6] Reinhart, R. C., Kacpura, T. J., Johnson, S. K., and Lux, J. P., "NASA's Space Communications and Navigation Test Bed aboard the International Space Station," *IEEE Aerospace and Electronic Systems Magazine*, Vol. 28, No. 4, 2013, pp. 4–15.

[7] Reinhart, R., and Lux, J. P., "Space-based Reconfigurable Software Defined Radio Test Bed aboard International Space Station," *SpaceOps 2014 conference*, 2014, p. 1612.

[8] NASA, "Space Communications and Navigation (SCaN) Network Service Catalog," Tech. rep., 2011.

[9] NASA, "Space Network User's Guide," Tech. rep., 2012.

[10] "Global Positioning Systems Directorate System Engineering & Integration Global Positioning System Interface Specification IS-GPS-200H," Tech. rep., 2013.

[11] Matsuoka, M., Rock, S. M., and Bualat, M. G., "Autonomous deployment of a self-calibrating pseudolite array for mars rover navigation," *Position Location and Navigation Symposium*, 2004, pp. 733–739.

[12] Anzalone, E. J., "Agent and model-based simulation framework for deep space navigation analysis and design," Ph.D. thesis, Georgia Institute of Technology, 2013.

[13] McComas, D., Strege, S., and Wilmot, J., "core Flight System (cFS) A Low Cost Solution for SmallSats," *29th Annual AIAA/USU Small Satellite Conference*, 2015.

[14] Blossom, E., "GNU radio: tools for exploring the radio frequency spectrum," *Linux Journal*, Vol. 2004, No. 122, 2004, p. 4.

[15] Acton Jr, C. H., "The SPICE concept-An approach to providing geometric and other ancillary information needed for interpretation of data returned from space science instruments," *AIAA and NASA, 2nd International Symposium on Space Information Systems*, 1990.

[16] Ossmann, M., "Software Defined Radio with HackRF," *Great Scott Gadgets, https://greatscottgadgets.com/sdr*, 2016.